Team Ade to Order -- Adrian Kloskowski, Kashf Mashrafi, Ying Xin Jiang

APCS2 pd8

Hw#56 -- Orienting Your Up Goer

2018-06-01

Scientific Calculator

Our final project aims to create a simulation of a physical scientific calculator, allowing the user to interact with a realistic array of button inputs to generate a mathematical expression to be calculated by a recursive tree-based function.

What It Uses:

- Processing: Provides a visual GUI mirroring buttons of a scientific calculator and its LED display.
- Object-Oriented Programing: The scientific calculator will be based around the abstract class Expression with a function getValue(). The value returned will be based on the designated function for the class implementing Expression, as well as any expressions contained within the class which represent parameters.
- Tree: The characters in the expression to be calculated are taken from a linear collection and arranged into a tree based on order of operations, with lower-priority expressions closer to the root. The value of the entire expression is then calculated recursively, as the value of each parent node is computed using the values of its children.

Priorities:

1. Solve simple single-operation problems.
2. Solve long problems using order of operations.
3. Solve exponents & squares, solve complex functions such as those used in trigonometry, and return the values of mathematical constants.
4. Create and implement an algorithm to map a linear collection of mathematical “characters” to an Expression tree.
5. Create algorithm to read an un-processed linear collection of characters and display it on a simulated LED screen, taking into account length of the characters and the user’s cursor.
6. Implement error handling (such as divide by zero, unmatched parentheses, etc.) and display it on the LED screen.
7. Implement storing and calling variables.
8. Solve statistical functions that take any number of parameters.